LISTING OF THE CLAIMS

The following listing, if entered, replaces all prior versions of the claims in the present application.

1. (**Currently Amended**) A method ~~**for communicating**~~ comprising:

controlling a user interface presented by a web browser, comprising[[:]]

establishing a persistent connection between the web browser and a web server**, wherein**

the persistent connection is maintained for a period of time;

causing the web browser to provide a wait request to the web server **via the persistent connection,** wherein,

the causing results in the web browser ~~**waits for**~~ **being ready to accept** an asynchronous message ~~**and is capable of concurrently performing other tasks;**~~ **from the web server,**

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~**enables the web server to**~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**;

~~**causing the web server to push the asynchronous message to the web browser in response to an incoming event, wherein the web browser presents a user interface change in response to the**~~

~~asynchronous message;~~

identifying a source of the asynchronous message, wherein

the source of the asynchronous message is the target process;

[[and]]

associating the wait request with the source, wherein

the associating identifies the web browser as a recipient of the asynchronous message**; and**

in response to the web server receiving an incoming event,

asynchronously pushing the asynchronous message from the web server to the web browser via the persistent connection, wherein

the asynchronous message corresponds to the incoming event, and

the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message.

2. (Cancelled)

3. (Cancelled)

4-5. (Cancelled)

6. (Previously Presented) The method of claim 1 further comprising:

generating the asynchronous message, the asynchronous message identifying the wait request, wherein the identifying identifies the web browser as a recipient of the asynchronous message; and

providing the asynchronous message to the web server.

7. (Original) The method of claim 6 wherein causing the web browser to provide the wait request comprises:

downloading requesting instructions to the web browser, wherein

the downloading causes the web browser to execute the requesting instructions.

8. (Original) The method of claim 6 further comprising:

storing a reference to a callback function with information from the wait request; and

using the reference to call the callback function when the asynchronous message is provided to the web server, wherein the callback function pushes the asynchronous message.

9. (Original) The method of claim 8 further comprising:

providing the callback function with context information, the context information identifying the web browser.

10. (Original) The method of claim 6 further comprising:

assigning the wait request to a connection between the web server and a business process server; and

listening to the connection for the asynchronous message.

11. (Original) The method of claim 6 further comprising:

assigning the wait request to a session between the web server and a business process server, the session being associated with a connection; and

listening to the connection for the asynchronous message for the session.

12. (Original) The method of claim 1 wherein causing the web server to push the asynchronous message comprises:

calling a callback function associated with the web browser when the asynchronous message is received, wherein the callback function pushes the asynchronous message.

13. (Original) The method of claim 12 further comprising:

storing a reference to the callback function; and

using the reference for calling the callback function.

14. (Original) The method of claim 13 further comprising:

storing a second reference to context information, the context information identifying the web browser; and

using the second reference for providing the context information to the callback

function.

15. (Previously Presented) The method of claim 1 wherein

the change in the user interface comprises at least one of a group consisting of the following:

causing a first user interface object to move to visually capture a user's attention;

causing a second user interface object to issue a sound to capture the user's attention;

presenting a screen pop of data; and

bringing a web browser window to the front of a screen.

16. **(Currently Amended)** A method ~~**for communicating**~~ comprising:

~~**causing a web server to push an asynchronous message to a web browser in response to an incoming event, wherein**~~

~~**the web browser is not blocked from receiving information from the web server while the web browser waits for the asynchronous message;**~~

~~**the incoming event comprises a request to establish communication with a user;**~~

~~**the web browser performs an action in response to the asynchronous message, and**~~

~~**the incoming event is received by a communication server;**~~

establishing a persistent connection between a web browser and a web server, wherein

the persistent connection is maintained for a period of time;

causing the web browser to provide a wait request to the web server **via the persistent connection,** wherein[[,]]

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a

task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request is associated with the web browser, and

the wait request ~~enables the web server to~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**;

identifying a source of the asynchronous message; **[[and]]**

associating the wait request with the source, wherein the associating identifies the web browser as a recipient of the asynchronous message**; and**

in response to the web server receiving an incoming event, asynchronously pushing the asynchronous message from the web server to the web browser via the persistent connection, wherein

the asynchronous message corresponds to the incoming event,

the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message,

the incoming event comprises a request to establish communication with a user,

the web browser performs an action in response to the asynchronous message, and

the incoming event is received by a communication server.

17. (Previously Presented) The method of claim 16 wherein

the asynchronous message includes an action instruction to cause the web browser to perform the action; and

the wait request further specifies a target process of a plurality of processes, wherein the processes are configured to generate asynchronous messages.

18. (Original) The method of claim 16 wherein the performing the action comprises performing at least one of a group consisting of the following:

causing a first user interface object to move to visually capture a user's attention;

causing a second user interface object to issue a sound to capture the user's attention;

presenting a screen pop of data; and

bringing a web browser window to front of screen.

19. (**Currently Amended**) A method ~~**for communicating**~~ comprising:

establishing a first connection between a web browser and a web server**, wherein the first connection is maintained for a period of time**;

establishing a second connection between the web server and a business process server;

controlling a user interface presented by the web browser ~~**comprising:**~~ **by**

registering the web browser with the business process server;

~~**providing the web server with the asynchronous message to push to the web browser, the providing being performed by the business process server and the providing being performed in response to an incoming event, wherein**~~

~~**the web browser waits for the asynchronous message, and**~~

~~**the incoming event comprises a request to establish communication with a user; and**~~

~~**causing the web server to push the asynchronous message to the web browser, wherein the web browser performs a user interface change in response to the asynchronous message;**~~

causing the web browser to provide a wait request to the web server **via the first connection,** wherein,

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~**enables the web server to**~~ **facilitates** push**ing** the

asynchronous message **from the web server** to the web browser **via the first connection**;

identifying a source of the asynchronous message, wherein the source of the asynchronous message is the target process; and

associating the wait request with the source, wherein the associating identifies the web browser as a recipient of the asynchronous message**;**

providing the web server with the asynchronous message, wherein the providing is performed by the business process server, the providing is performed in response to an incoming event, the asynchronous message corresponds to the incoming event, and the incoming event comprises a request to establish communication with a user; and

asynchronously pushing the asynchronous message from the web server to the web browser via the first connection, wherein the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message.

20. (**Currently Amended**) A method ~~**for communicating**~~ comprising:

controlling a user interface presented by a web browser**,** comprising**[[:]]**

establishing a persistent connection between the web browser and a web server, wherein the persistent connection is maintained for a period of time;

registering the web browser as available to receive an asynchronous message~~**, wherein the web browser waits for the asynchronous message and is capable of concurrently performing other tasks, and the web browser is not blocked waiting for the asynchronous message**~~;

~~**causing a web server to push the asynchronous message to the web browser in response to an incoming event, wherein the incoming event comprises a request to establish communication with a user;**~~

~~the web browser presents a user interface change in response to the asynchronous message, and~~

~~the incoming event is received by a communication server;~~

causing the web browser to provide a wait request to the web server **via the persistent connection,** wherein,

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~enables the web server to~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**;

identifying a source of the asynchronous message, wherein

the source of the asynchronous message is the target process;

[[and]]

associating the wait request with the source, wherein the associating identifies the web browser as a recipient of the asynchronous message**;**

in response to the web server receiving an incoming event,

asynchronously pushing the asynchronous message from the web server to the web browser via the persistent connection, wherein

the incoming event comprises a request to establish communication with a user,

the incoming event is received by a communication server,

the asynchronous message corresponds to the incoming event, and

the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message.

21. (**Currently Amended**) A method ~~**for communicating**~~ comprising:

controlling a user interface presented by a web browser, comprising[[:]]

establishing a persistent connection between the web browser and a web server, wherein

the persistent connection is maintained for a period of time;

causing the web browser to provide a wait request to the web server **via the persistent connection,** wherein[[,]]

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~**enables the web server to**~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**;

identifying a source of the asynchronous message, wherein

the source of the asynchronous message is the target process; and

associating the wait request with the source, wherein the associating identifies the web browser as a recipient of the asynchronous message;

in response to the web server receiving an incoming event,

asynchronously pushing the asynchronous message **from the web server** to the web browser ~~**in response to an incoming event**~~ **via the persistent connection, wherein**

the asynchronous message corresponds to the incoming event, and

the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message,

~~**the web browser waits for the asynchronous message and is capable of concurrently performing other tasks;**~~

the incoming event comprises a request to establish communication with a user~~;~~

~~**the browser presents a user interface change in response to the asynchronous message**~~, and

the incoming event is received by a communication server.

22. (**Currently Amended**) A method ~~**for communicating**~~ comprising:

controlling a user interface presented by a web browser**,** comprising**[[:]]**

establishing a persistent connection between the web browser and a web server, wherein

the persistent connection is maintained for a period of time;

causing the web browser to provide a wait request to **[[a]]** **the** web server **via the persistent connection,** wherein,

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the

asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~**enables the web server to**~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**;

identifying a source of the asynchronous message, wherein

the source of the asynchronous message is the target process;

[[and]]

generating the asynchronous message, the asynchronous message identifying the web browser as a recipient of the asynchronous message, the generating being performed by the target;

providing the asynchronous message to the web server; and

~~**causing the web server to push**~~ **in response to the web server receiving an incoming event, asynchronously pushing** the asynchronous message **from the web server** to the web browser ~~**in response to an incoming event**~~ **via the persistent connection, wherein**

~~**the web browser is not blocked from receiving information from the web server while the web browser waits for the asynchronous message;**~~

the asynchronous message corresponds to the incoming event, and

the incoming event comprises a request to establish communication with a user;

the asynchronous message causes the web browser ~~**presents**~~ **to present** a user interface change**,** in response to the **web browser receiving the** asynchronous message**[[;]],** and

the incoming event is received by a communication server.

23. (**Currently Amended**) A computer program product comprising:

controlling instructions to control a user interface presented by a web browser, comprising[[:]]

establishing instructions to cause a web server to establish a persistent connection between the web browser and the web server, wherein

the persistent connection is maintained for a period of time;

~~**pushing instructions to cause a web server to push an asynchronous message to the web browser in response to an incoming event, wherein**~~

~~**the web browser waits for the asynchronous message and is capable of concurrently performing other tasks;**~~

~~**the incoming event comprises a request to establish communication with a user;**~~

~~**the web browser presents a user interface change in response to the asynchronous message, and**~~

~~**the incoming event is received by a communication server;**~~

providing instructions to cause the web browser to provide a wait request to the web server **via the persistent connection,** wherein[[,]]

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~**enables the web server to**~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web

browser **via the persistent connection**;

identifying instructions to identify a source of the asynchronous message, wherein

the source of the asynchronous message is the target process; and

associating instructions to associate the wait request with the source,

wherein the associating identifies the web browser as a recipient of the asynchronous message;

pushing instructions to cause a web server to asynchronously push the asynchronous message from the web server to the web browser via the persistent connection, in response to an incoming event, wherein

the asynchronous message corresponds to the incoming event, and

the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message,

the incoming event comprises a request to establish communication with a user, and

the incoming event is received by a communication server; and

a **non-transitory** computer-readable **storage** medium for storing the controlling instructions, the **~~pushing~~ establishing** instructions, the providing instructions, the identifying instructions, **[[and]]** the associating instructions**, and the pushing instructions**.

24. (Cancelled)

25. (**Currently Amended**) The computer program product of claim 23 further comprising:

request providing instructions to cause the web browser to provide a wait request to the web server, the wait request being associated with the web browser;

generating instructions to generate the asynchronous message, the asynchronous message identifying the wait request, wherein the identifying identifies the web browser as a recipient of the asynchronous message; and

message providing instructions to provide the asynchronous message to the web server;

wherein the **non-transitory** computer-readable **storage** medium further stores the request providing instructions, the generating instructions, and the message providing instructions.

26. (**Currently Amended**) The computer program product of claim 25 further comprising:

storing instructions to store a reference to a callback function with information from the wait request; and

using instructions to use the reference to call the callback function when the asynchronous message is provided to the web server, wherein the callback function pushes the asynchronous message;

wherein the **non-transitory** computer-readable **storage** medium further stores the storing instructions and the using instructions.

27. (**Currently Amended**) The computer program product of claim 26 further comprising:

context providing instructions to provide the callback function with context information, the context information identifying the web browser;

wherein the **non-transitory** computer-readable **storage** medium further stores the context providing instructions.

28. (**Currently Amended**) The computer program product of claim 25 further comprising:

assigning instructions to assign the wait request to a connection between the web server and a business process server; and

listening instructions to listen to the connection for the asynchronous message;

wherein the **non-transitory** computer-readable **storage** medium further stores the assigning instructions and the listening instructions.

29. (**Currently Amended**) The computer program product of claim 23 wherein the pushing instructions comprise:

calling instructions to call a callback function associated with the web

browser when the asynchronous message is received, wherein the callback function pushes the asynchronous message; and

the **non-transitory** computer-readable **storage** medium further stores the calling instructions.

30. (**Currently Amended**) The computer program product of claim 29 further comprising:

reference storing instructions to store a reference to the callback function; and

reference using instructions to use the reference for calling the callback function;

wherein the **non-transitory** computer-readable **storage** medium further stores the reference storing instructions and the reference using instructions.

31. (**Currently Amended**) The computer program product of claim 30 further comprising:

context storing instructions to store a second reference to context information, the context information identifying the web browser; and

context using instructions to use the second reference for providing the context information to the callback function;

wherein the **non-transitory** computer-readable **storage** medium further stores the context storing instructions and the context using instructions.

32. (**Currently Amended**) The computer program product of claim 23 further comprising:

user interface changing instructions configured to perform at least one of a group consisting of the following:

cause a first user interface object to move to visually capture a user's attention;

cause a second user interface object to issue a sound to capture the user's attention;

present a screen pop of data; and

bring a web browser window to the front of a screen;

wherein the **non-transitory** computer-readable **storage** medium further stores the user interface changing instructions.

33. (**Currently Amended**) A computer program product comprising:

controlling instructions to control a user interface presented by a web browser, comprising[[:]]

establishing instructions to establish a persistent connection between the web browser and a web server, wherein

the persistent connection is maintained for a period of time;

registering instructions to register the web browser as available to receive an asynchronous message~~, wherein~~

~~the web browser waits for the asynchronous message and is capable of concurrently performing other tasks; and~~

~~the web browser is not blocked waiting for the asynchronous message~~;

~~pushing instructions to cause a web server to push the asynchronous message to the web browser in response to an incoming event, wherein~~

~~the incoming event comprises a request to establish communication with a user;~~

~~the web browser presents a user interface change in response to the asynchronous message, and~~

~~the incoming event is received by a communication server;~~

providing instructions to cause the web browser to provide a wait request to the web server **via the persistent connection,** wherein[[,]]

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~enables the web server to~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**;

identifying instructions to identify a source of the asynchronous message**,** wherein

the source of the asynchronous message is the target process; and

associating instructions to associate the wait request with the source,

wherein the associating identifies the web browser as a recipient of the asynchronous message;

pushing instructions to cause a web server to asynchronously push the asynchronous message from the web server to the web browser via the persistent connection, in response to an incoming event, wherein

the asynchronous message corresponds to the incoming event, and

the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message,

the incoming event comprises a request to establish communication with a user,

the incoming event is received by a communication server; and

a **non-transitory** computer-readable **storage** medium for storing the controlling instructions, **the establishing instructions,** the registering instructions, ~~**the pushing instructions,**~~ the providing instructions, the identifying instructions, **[[and]]** the associating instructions**, and the pushing instructions**.

34. **(Currently Amended)** A computer system comprising:

a processor;

a memory, the memory storing instructions for executing on the processor, the instructions comprising:

controlling instructions to control a user interface presented by a web

browser, comprising[[:]]

establishing instructions to establish a persistent connection between the web browser and a web server, wherein the persistent connection is maintained for a period of time;

~~**pushing instructions to cause a web server to push an asynchronous message to the web browser in response to an incoming event, wherein**~~

~~**the incoming event comprises a request to establish communication with a user,**~~

~~**the web browser waits for the asynchronous message and is capable of concurrently performing other tasks;**~~

~~**the web browser presents a user interface change in response to the asynchronous message, and**~~

~~**the incoming event is received by a communication server;**~~

providing instructions to cause the web browser to provide a wait request to the web server **via the persistent connection,** wherein[[,]]

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~enables the web server to~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**;

identifying instructions to identify a source of the asynchronous message, wherein

the source of the asynchronous message is the target process; and

associating instructions to associate the wait request with the source, wherein the associating identifies the web browser as a recipient of the asynchronous message**; and**

pushing instructions to cause a web server to asynchronously push the asynchronous message from the web server to the web browser via the persistent connection, in response to an incoming event, wherein

the asynchronous message corresponds to the incoming event, and

the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message,

the incoming event comprises a request to establish communication with a user,

the incoming event is received by a communication server.

35. (Cancelled)

36. (Original) The computer system of claim 34 wherein the instructions further comprise:

request providing instructions to cause the web browser to provide a wait request to the web server, the wait request being associated with the web browser;

generating instructions to generate the asynchronous message, the asynchronous message identifying the wait request, wherein the identifying identifies the web browser as a recipient of the asynchronous message; and

message providing instructions to provide the asynchronous message to the web server.

37. (Original) The computer system of claim 36 wherein the instructions further comprise:

storing instructions to store a reference to a callback function with information from the wait request; and

using instructions to use the reference to call the callback function when the asynchronous message is provided to the web server, wherein the callback function pushes the asynchronous message.

38. (Original) The computer system of claim 37 wherein the instructions further comprise:

context providing instructions to provide the callback function with context information, the context information identifying the web browser.

39. (Original) The computer system of claim 36 wherein the instructions further comprise:

assigning instructions to assign the wait request to a connection between the web server and a business process server; and

listening instructions to listen to the connection for the asynchronous message.

40. (Original) The computer system of claim 34 wherein the pushing instructions comprise:

calling instructions to call a callback function associated with the web browser when the asynchronous message is received, wherein the callback function pushes the asynchronous message.

41. (Original) The computer system of claim 40 wherein the instructions further comprise:

reference storing instructions to store a reference to the callback function; and

reference using instructions to use the reference for calling the callback function.

42. (Original) The computer system of claim 41 wherein the instructions further comprise:

context storing instructions to store a second reference to context information, the context information identifying the web browser; and

context using instructions to use the second reference for providing the context information to the callback function.

43. (Previously Presented) The computer system of claim 34 wherein the instructions further comprise:

user interface changing instructions configured to perform at least one of a group consisting of the following:

cause a first user interface object to move to visually capture a user's attention;

cause a second user interface object to issue a sound to capture the user's attention;

present a screen pop of data; and

bring a web browser window to the front of a screen.

44. (**Currently Amended**) A computer system comprising:

a processor;

a memory, the memory storing instructions for executing on the processor, the instructions comprising:

controlling instructions to control a user interface presented by a web browser, comprising[[:]]

establishing instructions to establish a persistent connection between the web browser and a web server, wherein the persistent connection is maintained for a period of time;

registering instructions to register the web browser as available to

receive an asynchronous message~~, wherein~~

~~the web browser waits for the asynchronous message and is capable of concurrently performing other tasks, and~~

~~the web browser is not blocked waiting for the asynchronous message~~;

~~pushing instructions to cause a web server to push the asynchronous message to the web browser in response to an incoming event, wherein~~

~~the incoming event comprises a request to establish communication with a user;~~

~~the web browser presents a user interface change in response to the asynchronous message, and~~

~~the incoming event is received by a communication server.~~

providing instructions to cause the web browser to provide a wait request to the web server **via the persistent connection,** wherein[[,]]

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~enables the web server to~~ **facilitates** push**ing** the asynchronous message **from the web**

server to the web browser **via the persistent connection**;

identifying instructions to identify a source of the asynchronous message**,** wherein

the source of the asynchronous message is the target process; **[[and]]**

associating instructions to associate the wait request with the source, wherein

the associating identifies the web browser as a recipient of the asynchronous message**; and**

pushing instructions to cause a web server to asynchronously push the asynchronous message from the web server to the web browser via the persistent connection, in response to an incoming event, wherein

the asynchronous message corresponds to the incoming event, and

the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message,

the incoming event comprises a request to establish communication with a user,

the incoming event is received by a communication server.

45. (**Currently Amended**) A system comprising:

a client computer comprising:

a web browser, wherein the web browser presents a user interface;

a **web** server computer coupled to the client computer, wherein the **web** server computer comprises

controlling means for controlling the user interface presented by the web browser,

establishing means for establishing a persistent connection between the web browser and the web server, wherein the persistent connection is maintained for a period of time;

~~**pushing means for causing a web server to push an asynchronous message to the web browser in response to an incoming event, wherein**~~

~~**the web browser waits for the asynchronous message and is capable of concurrently performing other tasks;**~~

~~**the incoming event comprises a request to establish communication with a user;**~~

~~**the web browser presents a user interface change in response to the asynchronous message, and**~~

~~**the incoming event is received by a communication server,**~~

identifying means for identifying a source of the asynchronous message, wherein

the source of the asynchronous message is a target process of a plurality of processes, and

the processes are configured to generate asynchronous messages,

associating means for associating a wait request with the source, wherein

the associating identifies the web browser as a recipient of the asynchronous message, and

pushing means for causing the web server to asynchronously push the asynchronous message from the web server to the web browser via the persistent connection, in response to an incoming event, wherein

the asynchronous message corresponds to the incoming event, and

the asynchronous message causes the web browser to

present a user interface change, in response to the web browser receiving the asynchronous message,

the incoming event comprises a request to establish communication with a user;

the incoming event is received by a communication server,

the client computer comprises

providing means for causing the web browser to provide

the wait request to the web server **via the persistent connection,** wherein,

the causing results in the web browser being ready to accept an asynchronous message from the server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies the target process,

the wait request is associated with the web browser, and

the wait request ~~**enables the web server to**~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**;

46. (Cancelled)

47. (**Currently Amended**) The system of claim 45, the **web** server computer further comprising:

generating means for generating the asynchronous message, the asynchronous message identifying the wait request, wherein the identifying identifies the web browser as a recipient of the asynchronous message; and

message providing means for providing the asynchronous message to the web server.

48. (**Currently Amended**) The system of claim 47, the **web** server computer further comprising:

storing means for storing a reference to a callback function with information from the wait request; and

using means for using the reference to call the callback function when the asynchronous message is provided to the web server, wherein the callback function pushes the asynchronous message.

49. (Previously Presented) The system of claim 48, the client computer further comprising:

context providing means for providing the callback function with context information, the context information identifying the web browser.

50. (Previously Presented) The system of claim 47, the server computer further comprising:

assigning means for assigning the wait request to a connection between the web server and a business process server; and

listening means for listening to the connection for the asynchronous message.

51. (Original) The system of claim 45 wherein the pushing means comprise:

calling means for calling a callback function associated with the web browser when the asynchronous message is received, wherein the callback function pushes the asynchronous message.

52. (**Currently Amended**) The system of claim 51, the **web** server computer further comprising:

reference storing means for storing a reference to the callback function; and

reference using means for using the reference for calling the callback function.

53. (**Currently Amended**) The system of claim 52, the **web** server computer further comprising:

context storing means for storing a second reference to context information, the context information identifying the web browser; and

context using means for using the second reference for providing the context information to the callback function.

54. (Previously Presented) The system of claim 45, the client computer further comprising:

user interface changing means configured to perform at least one of a group consisting of the following:

cause a first user interface object to move to visually capture a user's attention;

cause a second user interface object to issue a sound to capture the user's attention;

present a screen pop of data; and

bring a web browser window to front of screen.

55. (**Currently Amended**) A system comprising:

a client computer comprising:

a web browser, wherein the web browser presents a user interface;

a **web** server computer coupled to the client computer, wherein the **web** server computer comprises

controlling means for controlling a user interface presented by a web browser,

establishing means for establishing a persistent connection between the web browser and a web server, wherein the persistent connection is maintained for a period of time;

registering means for registering the web browser as available to receive an asynchronous message~~, wherein the web browser waits for the asynchronous message and is capable of concurrently performing other tasks, and the web browser is not blocked waiting for the asynchronous~~

~~message, and;~~

~~pushing means for causing a web server to push the asynchronous message to the web browser in response to an incoming event, wherein~~

~~the incoming event comprises a request to establish communication with a user;~~

~~the web browser presents a user interface change in response to the asynchronous message, and~~

~~the incoming event is received by a communication server,~~

identifying means for identifying a source of the asynchronous message, wherein

the source of the asynchronous message is a target process of a plurality of processes, and

the processes are configured to generate asynchronous messages,

associating means for associating a wait request with the source, wherein

the associating identifies the web browser as a recipient of the asynchronous message,

pushing means for asynchronously pushing the asynchronous message from the web server to the web browser via the persistent connection, in response to the web server receiving an incoming event, wherein

the asynchronous message corresponds to the incoming event, and

the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message, and

the client computer comprises

providing means for causing the web browser to provide the wait request to the web server **via the persistent connection**, wherein

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies the target process,

the wait request is associated with the web browser, and

the wait request ~~enables the web server to~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**.

56. (Cancelled)

57. (Cancelled)

58. (**Currently Amended**) A system comprising:

a controlling module to control a user interface presented by a web browser ~~comprising:~~

~~a pushing module to cause a web server to push an asynchronous message to the web browser in response to an incoming event, wherein~~

~~the incoming event comprises a request to establish communication with a user;~~

~~the web browser presents a user interface change in response to the asynchronous message, and~~

~~the incoming event is received by a communication server~~;

an establishing module to establish a persistent connection between the web browser and a web server, wherein

the persistent connection is maintained for a period of time;

a request providing module to cause the web browser to provide a wait request to the web server **via the persistent connection,** wherein,

the causing results in the web browser being ready to accept an asynchronous message from the web server,

the asynchronous message is asynchronously pushed from the web server,

the providing the wait request allows the web browser to perform a task other than awaiting receipt of the asynchronous message, at least during the period of time,

the wait request specifies a target process of a plurality of processes,

the processes are configured to generate asynchronous messages,

the wait request is associated with the web browser, and

the wait request ~~**enables the web server to**~~ **facilitates** push**ing** the asynchronous message **from the web server** to the web browser **via the persistent connection**;

an identifying module to identify a source of the asynchronous message wherein the source of the asynchronous message is the target process;

an associating module to associate the wait request with the source, wherein the associating identifies the web browser as a recipient of the asynchronous message;

a pushing module to asynchronously push the asynchronous message from the web server to the web browser via the persistent connection, in response to the web server receiving an incoming event, wherein the asynchronous message corresponds to the incoming event, and the asynchronous message causes the web browser to present a user interface change, in response to the web browser receiving the asynchronous message,

the incoming event comprises a request to establish communication with a user, and

the incoming event is received by a communication server; and

a **non-transitory** computer-readable storage medium configured to store the controlling module, ~~**pushing**~~ **establishing** module, request providing module, identifying module, **[[and]]** associating module**, and pushing module**.

59. (Cancelled)

60. (**Currently Amended**) The system of claim 58 further comprising:

a generating module to generate the asynchronous message, the asynchronous message identifying the wait request, wherein the identifying identifies the web browser as a recipient of the asynchronous message; and

a message providing module to provide the asynchronous message to the web server, wherein

the **non-transitory** computer-readable storage medium is configured to store the generating module and message providing module.

61. (**Currently Amended**) The system of claim 60 further comprising:

a storing module to store a reference to a callback function with information from the wait request; and

a using module to use the reference to call the callback function when the asynchronous message is provided to the web server, wherein the callback function pushes the asynchronous message, wherein

the **non-transitory** computer-readable storage medium is configured to store the storing module and using module.

62. (**Currently Amended**) The system of claim 61 further comprising:

a context providing module to provide the callback function with context information, the context information identifying the web browser, wherein

the **non-transitory** computer readable storage medium is configured to store the context providing module.

63. (**Currently Amended**) The system of claim 60 further comprising:

an assigning module to assign the wait request to a connection between the web server and a business process server; and

a listening module to listen to the connection for the asynchronous message, wherein

the **non-transitory** computer readable storage medium is configured to store the assigning module and listening module.

64. (**Currently Amended**) The system of claim 58 wherein the pushing means comprise:

a calling module to call a callback function associated with the web browser when the asynchronous message is received, wherein the callback function pushes the asynchronous message, wherein

the **non-transitory** computer readable storage medium is configured to store the calling module.

65. (**Currently Amended**) The system of claim 64 further comprising:

a reference storing module to store a reference to the callback function; and

a reference using module to use the reference for calling the callback function, wherein

the **non-transitory** computer readable storage medium stores the reference storing module and the reference using module.

66. (**Currently Amended**) The system of claim 65 further comprising:

a context storing module to store a second reference to context information, the context information identifying the web browser; and

a context using module to use the second reference for providing the context information to the callback function, wherein

the **non-transitory** computer readable storage medium stores the context storing module and the context using module.

67. (**Currently Amended**) The system of claim 58 further comprising:

a user interface changing module configured to perform at least one of a group consisting of the following:

cause a first user interface object to move to visually capture a user's attention;

cause a second user interface object to issue a sound to capture the user's attention;

present a screen pop of data; and

bring a web browser window to front of screen, wherein

the **non-transitory** computer readable storage medium is configured to store the user interface changing module.

68. (Previously Presented) The method of claim 1 wherein the persistent connection comprises a hypertext transfer protocol (HTTP) connection between the web browser and the web server when a user logs in.

69. (Previously Presented) The method of claim 1 further comprising:
storing the wait request in memory;
removing the wait request from memory in response to pushing the asynchronous message.

70. (**New**) The method of claim 1, wherein the persistent connection is closed when the asynchronous message is asynchronously pushed to the web browser.

71. (**New**) The method of claim 1, wherein the persistent connection persistent after the asynchronous message is asynchronously pushed to the web browser.